

PATENT

1

Docket No.24,723 USA

METHOD AND APPARATUS FOR SERVER SIDE QUEUING TO CONTROL PAGE
PRESENTATION ON WEB ENABLED DEVICES

Field of the Invention

5 The invention pertains to Web servers. More particularly, the invention pertains to software for controlling the order of serving of supplemental files that form a part of a requested Web page.

Background of the Invention

10 The World Wide Web (hereinafter the Web) is a world wide network of computers that are interconnected and able to communicate with each other via the Internet. Hypertext transfer protocol(http) is the protocol used for transferring Web pages and supplemental files over the Internet. Web servers are computers that form part of the Web and whose
15 general purpose is to provide information to other computers coupled to the Web. Those computers that are primarily used

to request and receive information via the Web from Web servers are typically termed client machines or client computers.

On the Web, information is served in the form of Web pages written in HTML (HyperText Markup Language). Thus, for example, a retail Web sight operator couples to the internet via a server on which is stored a plurality of Web pages written in HTML programming language. The HTML defining a Web page defines the manner of presentation of information on the client machine. The HTML code also typically includes the textual content of the page. However, other types of content, such as images, audio, background, and multimedia are contained in separate, supplemental, files stored on the server which are referenced within the HTML code by HTML tags. For instance, images may be stored in separate files in jpeg format or gif format. Audio files may be stored in mp3, wav, wma(the Windows™ media application), or other formats.

In a common type of example, a Web retailer has a server coupled to the Web. A customer accesses the Web retailer's Web site from a desktop computer using Web browser software such as Microsoft Internet Explorer™ or Netscape Navigator™. The customer's desktop PC utilizing the Web browser software would be considered a client machine.

The Web browser running on the client machine requests a particular Web page in a manner well known to those of skill in the art. Upon receipt of the request for a particular Web

page, the server serves the HTML code for that page to the client machine via the Internet.

As noted above, the HTML code for any given page is likely to contain a references to supplemental files such as image files, audio files and/or multimedia files. When the browser receives the HTML code for the page, it parses the page to find references to supplemental files and sends requests back to the server for the supplemental files referenced in the code.

Browsers typically read the HTML code in a standard left to right, top to bottom manner and send out requests for the referenced supplemental files in the order in which they are encountered while reading the page. The server receives the requests and sends the supplemental files back to the browser in the order in which the requests are received.

Typically, most browsers support multiple virtual ports for receiving supplemental files. Thus, for example, a browser with support for four virtual ports can send out four requests for supplemental files and begin receiving those supplemental files simultaneously. When a supplemental file is received in full, the browser will send out another request for the next supplemental file (assuming there are five or more supplemental files) until all of the supplemental files in the page are fully received and rendered by the browser. Since some files may be smaller than others, the order in which the requests for supplemental files are sent out does not

necessarily correspond to the order to which they are received in their entirety. For instance, a small file that is requested after a larger file may be completely received and rendered before the earlier larger file.

5 Accordingly, the content of a page is rendered in a piece meal fashion dictated primarily by the order in which the supplemental files are referenced in the HTML code as well as the size of the files.

10 Some supplemental files associated with a Web page may require a significant period to download, during which time the individual at the client machine may become impatient. For instance, high resolution and/or large image files, audio files, multimedia files and video files can take a large amount of time to download. During that time, the user at the client machine may become bored or impatient. Thus, it is desirable to be able to have one or more portions of the content of the page be downloaded and displayed first so that the user at the client machine will have something to occupy his or her attention while the remainder of the supplemental files download.

20 Therefore, it is an object of the present invention to provide an improved method and apparatus for downloading Web pages, including supplemental files.

25 It is another object of the present invention to provide a method and apparatus for controlling the order in which

supplemental files referenced in a Web page are served to the client machine.

It is yet a further object of the present invention to provide a method and apparatus by which the server controls the order in which supplemental files referenced in a HTML page are served to the client machine.

Summary Of The Invention

In accordance with the method and apparatus of the present invention, the order in which supplemental files referenced in a Web page are downloaded from the server to a requesting client is specified by the designer in the HTML code of the Web page and controlled at the server-side.

In accordance with the invention, regardless of the order in which the client-side Web browser requests supplemental files referenced in a received Web page, the server serves the supplemental files in a manner dictated by the Web page designer. More particularly, each supplemental file referenced in a Web page has a sequence number associated with it. In a preferred embodiment, the sequence number may be provided as an additional attribute of the tag associated with the supplemental file. The sequence number attribute will not be understood by the browser software at the client-side and thus will generally be ignored by the browser software. However, at the server-side, when a page is requested, the server parses the page to find the sequence number attribute

associated with each supplemental file. It then builds a queue of supplemental files to be served to the client machine, the supplemental files being queued in the order dictated by the sequence number.

5 Thereafter, regardless of the order in which the browser returns requests for supplemental files, the supplemental files will be served in the order dictated in the queue. Existing browsers already are equipped to receive and cache files and associate such cached files with files referenced in an HTML page. Accordingly, the fact that the supplemental
10 files referenced in a Web page may be received in an order different than the order in which the browser requested them generally is of no consequence. The browser generally still will render the supplemental files upon their being fully
15 received. Accordingly, the invention resides entirely at the server side and will work with widely available Web browsers.

Brief Description of the Drawings

Figure 1 is a screen capture of an exemplary, fully rendered, Web page.

20 Figure 2 is a screen capture of the exemplary Web page of Figure 1 that has been partially rendered in accordance with the prior art.

Figure 3 is a screen capture of the Web page of Figures 1 and 2 at a point in time in which it has been partially
25 rendered in accordance with the present invention.

Figure 4 is a flow diagram in accordance with the present invention.

Detailed Description of the Invention

Figure 1 is a screen capture of an exemplary Web page containing a plurality of supplemental files. Table 1 shows the HTML code for the page shown in Figure 1. The supplemental files include three thumbnail image files 12, 14 and 16 and two large, high resolution, .gif image files. It also includes an audio file that will play music while one views the page. The audio does not appear on the screen and therefore is not shown in Figure 1. Finally, the Web page includes a video file in MPEG format that runs automatically when the page is loaded (also not shown in Figure 1).

TABLE 1

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

<html>
<head>
  <title>Web Page Example</title>
</head>

<body>

<div align="center">


<br>
<table border="0" width="480">
  <tr>
    <td>This is a Web page designed to illustrate the invention disclosed and claimed in U.S.
      Patent Application No. 09/000,000. This text is just a bunch of nonsense that you
      should imagine has something to do with the images, video clip and sound clip contained
      in this page. Images 18 and 20 above are large, high resolution jpeg images that will
      each take a long time to download.</td>
  </tr>
</table>
</div>
```


<div align="center">
<embed src="video-clip24.mpg" width="320" height="240" controller="true" border="0"
cache="false">
</div>

<div align="center">
<embed src="audio-file.mid" volume="50" height="60" width="144" hidden="true">
</div>

<div align="center">

<table border="0" width="400">

<tr>
<td>However, images 12, 14, and 16 are small thumbnail images that will download and, thus,
be rendered much more quickly. The video clip and sound clip are really big files too
and would take a really long time to download. You can't see them, but trust me.
They're really big.</td>

</tr>
</table>
</div>

</body>
</html>

TABLE 1

As previously noted, in the prior art, the browser at the client-side will request the supplemental files in the order in which they are encountered as the browser software reads the code of the page. The browser software reads the HTML code generally in a left to right and top to bottom manner (the way a person would read a page of a book). Generally, browsers do not send out any requests for supplemental files until the HTML code of the page has been fully loaded at the client-side.

It can be seen from the HTML code of Table 1 that the browser will, therefore, encounter the reference to the two

large .gif files first, then the MPEG file, then the audio file and then the three thumbnail image files.

If we assume that this particular Web browser can have four virtual ports open simultaneously for downloading supplemental files, then it will download the two .gif files, the MPEG file and the audio file simultaneously and will not be able to begin downloading the much smaller thumbnail image files until one or more of the first four files completes downloading. Accordingly, after a certain amount of time, for example, 10 seconds, only the first supplemental file may be fully downloaded and the user of the Web browser may see the screen shown in Figure 2 after 10 seconds. If the thumbnails were allowed to download first, the user would at least be able to view the small, low resolution thumbnail images much more quickly while the larger images, MPEG and audio files were downloading.

The present invention provides a method and apparatus by which the designer of the Web page can cause the browser to download the thumbnail images before the other supplemental files even though they appear later in the HTML code of the page. Accordingly, when the Web page of Figures 1 and 2 is served to the Web browser by a server enabled in accordance with the method and apparatus of the present invention, the viewer at the client machine would be able to view the screen shown in Figure 3 almost instantaneously while the larger audio, MPEG and image files are still downloading.

In accordance with the invention, when the server receives a request for a Web page, it retrieves (or constructs) the page and, even before the page is sent to the requesting client, the server parses the page looking for references to supplemental files. In accordance with the invention, each reference to a supplemental file (which typically will take the form of an HTML tag of which one of the attributes is a file path) has associated therewith a sequence code indicating the order in which it should be served to the requesting client relative to all of the other supplemental files referenced in the page. In a preferred embodiment, the sequence code can simply be an additional attribute of the tag associated with the supplemental file. For instance, if the supplemental file is an image file referenced in an HTML tag, then the sequence number would simply be an additional attribute of the tag. For instance, an attribute seq, the value of which is an integer, e.g., seq = 1, 2, 3,..., could be used as an attribute of any and all HTML tags that refer to supplemental files, such as the <embed>, <object>, , and <applet> tags. Since "seq" has no defined meaning in HTML, when a browser receives a page with such an attribute, it simply will ignore it. However, a server enabled in accordance with the present invention will recognize the "seq =" attribute and read the numeral that appears after the equal sign as a number dictating the order in which that supplemental file should be sent to the

requesting Web browser relative to all other tags in the page having an assigned "seq" attribute. The server then builds a queue listing the names of the supplemental in the order dictated by their sequence number and serves those supplemental files in the order dictated by the queue. A default scheme can be readily developed for tags that reference supplemental files but for which a "seq" attribute has not been assigned, if any. For instance, such supplemental files can be served last, after the supplemental files with assigned sequence numbers have been served.

When the Web browser at the client-side receives the full page, it will send out requests for the supplemental files referenced within the HTML code of that page in a particular order dictated by the browser software design, which probably will be in the order in which they are encountered in reading the page from left to right and top to bottom. In most instances, that order will not match the order in which the server has queued the files. Upon receiving the request for the supplemental files, the server detects these request in order to confirm that the page has been received by the client machine and may utilize those requests for other purposes. However, the server software essentially ignores the requests received from the client machine for purposes of serving up the supplemental files.

By queuing up a supplemental files in the order dictated by the designer of the Web page rather than the order dictated

by the Web browser running on the client machine, the supplemental files of the Web page can be rendered at the client machine in an order that makes the most sense. No modifications to the Web browser software are necessary. The Web browsers simply ignore the unrecognizable HTML code that dictates the queuing sequence, i.e., the "seq" attribute in the preferred embodiment discussed above.

It would be a simple matter to modify most Web browsers to enable them to receive and properly handle supplemental files regardless of the order in which the Web browser requested them in accordance with the present invention.

Accordingly, the Web page designers may dictate the order in which supplemental files are served to a requesting browser without being constrained by the order in which the relevant tags appear in the code and without any need to modify the software of the Web browsers themselves.

Table 2 is exemplary HTML code for the Web page of Figures 1-3 in accordance with the present invention in which sequence number attributes, "seq" have been added to all of the tags that reference supplemental files. It can be seen from Table 2 that the Web page designer has dictated that the thumbnail images 12, 14, and 16 be served first, followed by the image 18, followed by the image 20, followed by the audio file (not shown in Figures 1-3) and finally followed by the MPEG file 24.

TABLE 2

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
```

```
<html>
```

```
<head>
```

```
  <title>Web Page Example</title>
```

```
</head>
```

```
<body>
```

```
  <div align="center">
```

```
    
```

```
    
```

```
    <br>
```

```
    <table border="0" width="480">
```

```
      <tr>
```

```
        <td>This is a Web page designed to illustrate the invention disclosed and claimed in U.S.
```

```
        Patent Application No. 09/000,000. This text is just a bunch of nonsense that you
        should imagine has something to do with the images, video clip and sound clip contained
        in this page. Images 18 and 20 above are large, high resolution jpeg images that will
        each take a long time to download.</td>
```

```
      </tr>
```

```
    </table>
```

```
  </div>
```

```
  <br>
```

```
  <div align="center">
```

```
    <embed src="video-clip24.mpg" width="320" height="240" controller="true" border="0" cache="false"
    seq="7">
```

```
  </div>
```

```
  <br>
```

```
  <div align="center">
```

```
    <embed src="audio-file.mid" volume="50" height="60" width="144" hidden="true" seq="6">
```

```
  </div>
```

```
  <div align="center">
```

```
    
```

```
    
```

```
    
```

```
    <br>
```

```
    <table border="0" width="400">
```

```
      <tr>
```

```
        <td>However, images 12, 14, and 16 are small thumbnail images that will download and, thus,
        be rendered much more quickly. The video clip and sound clip are really big files too
        and would take a really long time to download. You can't see them, but trust me.
```

```
        They're really big.</td>
```

```
      </tr>
```

```
    </table>
```

```
  </div>
```

</body>
</html>

TABLE 2

Figure 4 is a flow chart illustrating operation at the
5 server-side in accordance with the present invention. In step
401, the server detects a request for a Web page from a
client. In step 403, the server retrieves or builds the Web
page as the case may be (many Web sites do not store all Web
pages, but build them from information contained in a
10 plurality of related tables as requests are received). In
step 405, before sending the HTML code comprising the
requested Web page to the requesting client, the server parses
the page to determine the sequence order for serving
supplemental files. In the preferred embodiment described
15 above, it is a matter of detecting the "seq =" attributes
within the HTML code of the page.

In step 407, the server constructs and stores a queue
listing the supplemental files in the order indicated by the
"seq" attributes (or whatever other manner the order is
20 indicated in the HTML code). In step 409, the server sends
the Web page code to the requesting client. Between step 409
and the next step 411 in the flowchart, the requesting client
presumably receives the complete page and starts sending out
requests back to the server for the supplemental files.
25 Accordingly, in step 411, the server receives those requests.
The server does not use those requests, however, for queuing
up and serving supplemental files. It may, however, use the

request to confirm that the page has been received by the client machine (and possibly other purposes unrelated to the present invention). In step 413, the server begins retrieving the supplemental files in the order dictated by the queue.

5 Finally, in step 415, the server serves the supplemental files to the client in the order in which they are retrieved in accordance with the queue.

It should be understood by those skilled in the art that Figure 4 is a simplified block diagram and that certain of the steps may be looped through, carried out simultaneously, and/or interrupt driven. For instance, with respect to Web pages that contain more than just a few supplemental files, requests from the client machine for supplemental files will be received at different times. Further, the server will
10 simultaneously serve only so many supplemental files simultaneously as there are virtual ports that the browser supports. As each supplemental file is completely received at the client machine, the server can move on to the next supplemental file in the queue and begin serving it to the
15 client machine.
20

While the invention has been described above in connection with a Web server and a requesting client comprising a desktop computer, it should be understood by those of skill in the related arts that the invention can be
25 employed in connection with any server serving files to any

type of client machine, including any type of Web enabled device.

Having thus described a few particular embodiments of the invention, various alterations, modifications, and

5 improvements will readily occur to those skilled in the art.

Such alterations, modifications and improvements as are made obvious by this disclosure are intended to be part of this description though not expressly stated herein, and are intended to be within the spirit and scope of the invention.

10 Accordingly, the foregoing description is by way of example only, and not limiting. The invention is limited only as defined in the following claims and equivalents thereto.